**ΕΡΓΑΣΙΑ 5**

**ΣΤΟ ΜΑΘΗΜΑ: ΣΥΣΤΗΜΑΤΑ ΑΝΑΚΑΛΥΨΗΣ ΓΝΩΣΗΣ ΑΠΟ ΒΔ**

1. **Α) Κατηγοριοποιητής Naive Bayes**

Αρχικά πάμε και υπολογίζουμε μια φορά την πράιορ για την κατηγορία ΄΄τηλεόραση΄΄ και άλλη μια φορά για την κατηγορία ΄΄οχιΤηλεόραση΄΄. Ουσιαστικά πάμε και βρίσκουμε την συχνότητα της κάθε κατηγορίας στο training set.

**P(τηλεόραση) = 3/5 = 0.6**

**P(οχιΤηλεόραση) = 2/5 = 0.4**

Δηλαδή τρία στα πέντε έγγραφα ανήκουν στην κατηγορία ΄΄τηλεόραση΄΄ και τα άλλα δύο πέμπτα που μένουν ανήκουν στην κατηγορία ΄΄οχιΤηλεόραση΄΄.

Τώρα για κάθε μια λέξη που έχω, υπολογίζω την δεσμευμένη πιθανότητα και για τις δύο κατηγορίες.

P(πρόγραμμα|τηλεόραση) = (3+1) / (14+8) = 4/22 = 0.18

P(πρόγραμμα|οχιΤηλεόραση) = (0+1) / (6+8) = 1/14 = 0.07

P(επεισόδιο|τηλεόραση) = (5+1) / (14+8) = 6/22 = 0.27

P(επεισόδιο|οχιΤηλεόραση) = (1+1) / (6+8) = 2/14 = 0.14

P(σειρά|τηλεόραση) = (1+1) / (14+8) = 2/22 = 0.09

P(σειρά|οχιΤηλεόραση) = (0+1) / (6+8) = 1/14 = 0.07

P(κανάλι|τηλεόραση) = (3+1) / (14+8) = 4/22 = 0.18

P(κανάλι|οχιΤηλεόραση) = (0+1) / (6+8) = 1/14 = 0.07

P(ταινία|τηλεόραση) = (1+1) / (14+8) = 2/22 = 0.09

P(ταινία|οχιΤηλεόραση) = (0+1) / (6+8) = 1/14 = 0.0

P(ειδήσεις|τηλεόραση) = (1+1) / (14+8) = 2/22 = 0.09

P(ειδήσεις|οχιΤηλεόραση) = (1+1) / (6+8) = 2/14 = 0.14

P(γήπεδο|τηλεόραση) = (0+1) / (14+8) = 1/22 = 0.04

P(γήπεδο|οχιΤηλεόραση) = (2+1) / (6+8) = 3/14 = 0.21

P(ομάδα|τηλεόραση) = (0+1) / (14+8) = 1/22 = 0.04

P(ομάδα|οχιΤηλεόραση) = (2+1) / (6+8) = 3/14 = 0.21

**P(τηλεόραση|d6) = P(τηλεόραση) \* P(επεισόδιο|τηλεόραση)^2 \* P(γήπεδο|τηλεόραση)^3 \* P(ειδήσεις|τηλεόραση) = 3/5 \* (6/22)^2 \* (1/22)^3 \* 2/22 = 0.00000038**

**P(οχιΤηλεόραση|d6) = P(οχιΤηλεόραση) \* P(επεισόδιο|οχιΤηλεόραση)^2 \* P(γήπεδο|οχιΤηλεόραση)^3 \* P(ειδήσεις|οχιΤηλεόραση) = 2/5 \* (2/14)^2 \* (3/14)^3 \* 2/14 = 0.0000114**

**ΣΥΜΠΕΡΑΣΜΑ:** Η ποστεριόρι για την κατηγορία ΄΄οχιΤηλεόραση΄΄ είναι μεγαλύτερη άρα ο Classifier θα αποφασίσει ότι το έγγραφο μου θα ανήκει στην κατηγορία ΄΄οχιΤηλεόραση΄΄. Άρα θα έχουμε <<επεισόδιο επεισόδιο γήπεδο γήπεδο γήπεδο ειδήσεις 🡪 όχι >>.

**Β) Κατηγοριοποιητής Bernoulli Naive Bayes**

Ακολουθώντας την λογική του Α) ερωτήματος, υπολογίζουμε αυτή την φορά της πιθανότητες με τον τύπο του Bernoulli Naive Bayes.

**P(τηλεόραση) = 3/5 = 0.6**

**P(οχιΤηλεόραση) = 2/5 = 0.4**

P(πρόγραμμα|τηλεόραση) = (2+1) / (3+2) = 3/5 = 0.6

P(πρόγραμμα|οχιΤηλεόραση) = (0+1) / (2+2) = 1/4 = 0.25

P(επεισόδιο|τηλεόραση) = (3+1) / (3+2) = 4/5 = 0.8

P(επεισόδιο|οχιΤηλεόραση) = (1+1) / (2+2) = 2/4 = 0.5

P(σειρά|τηλεόραση) = (1+1) / (3+2) = 2/5 = 0.4

P(σειρά|οχιΤηλεόραση) = (0+1) / (2+2) = 1/4 = 0.25

P(κανάλι|τηλεόραση) = (2+1) / (3+2) = 3/5 = 0.6

P(κανάλι|οχιΤηλεόραση) = (0+1) / (2+2) = 1/4 = 0.25

P(ταινία|τηλεόραση) = (1+1) / (3+2) = 2/5 = 0.4

P(ταινία|οχιΤηλεόραση) = (0+1) / (2+2) = 1/4 = 0.25

P(ειδήσεις|τηλεόραση) = (1+1) / (3+2) = 2/5 = 0.4

P(ειδήσεις|οχιΤηλεόραση) = (1+1) / (2+2) = 2/4 = 0.5

P(γήπεδο|τηλεόραση) = (0+1) / (3+2) = 1/5 = 0.2

P(γήπεδο|οχιΤηλεόραση) = (2+1) / (2+2) = 3/4 = 0.75

P(ομάδα|τηλεόραση) = (0+1) / (3+2) = 1/5 = 0.2

P(ομάδα|οχιΤηλεόραση) = (2+1) / (2+2) = 3/4 = 0.75

**P(τηλεόραση|d6) = P(τηλεόραση) \* P(επεισόδιο|τηλεόραση) \* P(γήπεδο|τηλεόραση) \* P(ειδήσεις|τηλεόραση) \* (1-P(πρόγραμμα|τηλεόραση)) \* (1-P(σειρά|τηλεόραση)) \* (1-P(κανάλι|τηλεόραση)) \* (1-P(ταινία|τηλεόραση)) \* (1- P(ομάδα|τηλεόραση)) = 3/5 \* (4/5) \* (1/5) \* (2/5) \* (1-3/5) \* (1-2/5) \* (1-3/5) \* (1-2/5) \* (1-1/5) = 0.0017**

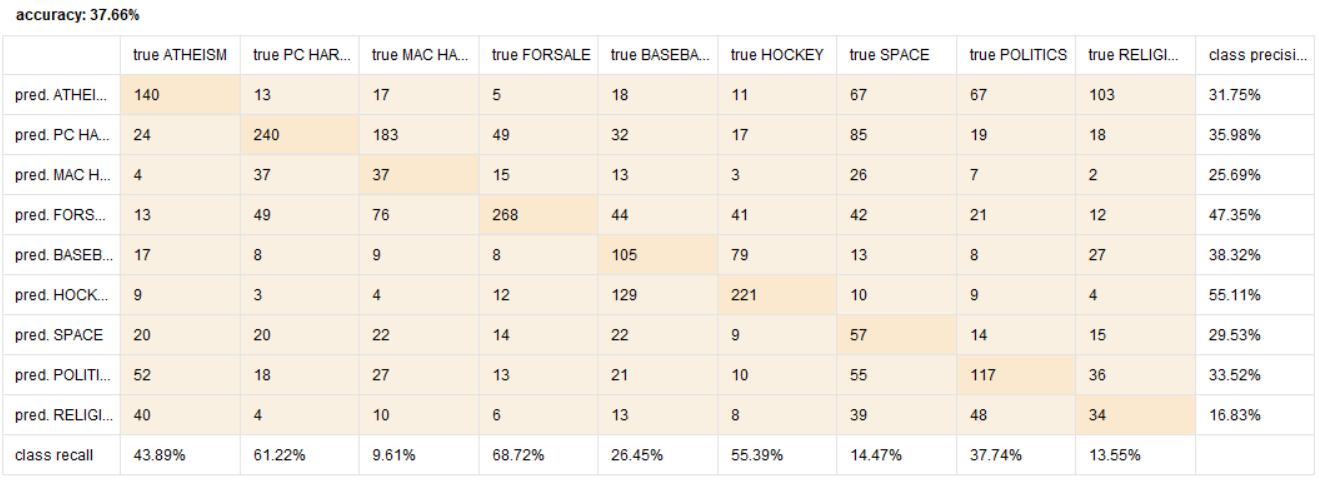
**P(οχιΤηλεόραση|d6) = P(οχιΤηλεόραση) \* P(επεισόδιο|οχιΤηλεόραση) \* P(γήπεδο|οχιΤηλεόραση) \* P(ειδήσεις|οχιΤηλεόραση) \* (1-P(πρόγραμμα|οχιΤηλεόραση)) \* (1-P(σειρά|οχιΤηλεόραση)) \* (1-P(κανάλι|οχιΤηλεόραση)) \* (1-P(ταινία|οχιΤηλεόραση)) \* (1- P(ομάδα|οχιΤηλεόραση)) = 2/5 \* (2/4) \* (3/4) \* (2/4) \* (1-1/4) \* (1-1/4) \* (1-1/4) \* (1-1/4) \* (1-3/4) = 0.0059**

**ΣΥΜΠΕΡΑΣΜΑ:** Η ποστεριόρι για την κατηγορία ΄΄οχιΤηλεόραση΄΄ είναι μεγαλύτερη άρα ο Classifier θα αποφασίσει ότι το έγγραφο μου θα ανήκει στην κατηγορία ΄΄οχιΤηλεόραση΄΄. Άρα θα έχουμε <<επεισόδιο επεισόδιο γήπεδο γήπεδο γήπεδο ειδήσεις 🡪 όχι >>.

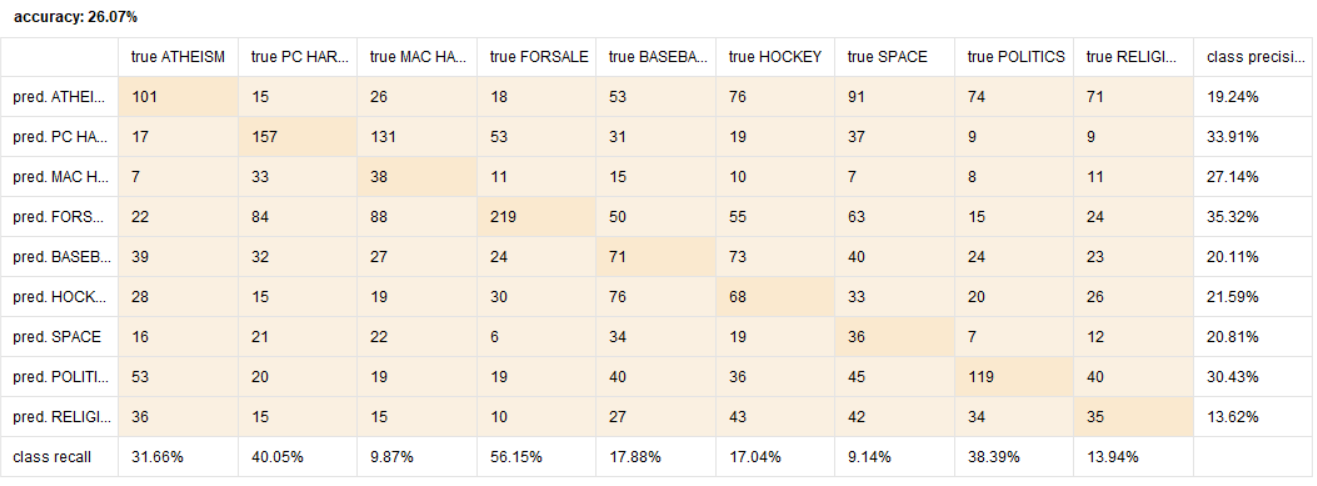
1. Αρχικά κατέβασα τα δεδομένα μου από τον ιστότοπο **qwone.com** και αφού επέλεξα την κατάλληλη έκδοση, πήγα στην συνέχεια και εισήγαγα τα στοιχειά μου στο RapidMiner. Έτρεξα 12 διαφορετικά πειράματα με σταθερή την προεπεξεργασία στα δεδομένα μου. Χρησιμοποίησα **Tokenization** για να κόψει τα στοιχειά μου σε γλωσσικές μονάδες, επίσης χρησιμοποίησα **Filter Stopwards (English)** ώστε να αφαιρέσει τις αγγλικές λέξεις κλειδιά, τοποθέτησα **Filter Tokens (by Length)** ώστε να διατηρήσει από 4 έως και 12 χαρακτήρες σε κάθε λέξη, επιπλέον χρησιμοποίησα τον **αλγόριθμο αποκοπής Porter** μιας και είναι ο πιο γνωστός και ευρεία διαδεδομένος. Τέλος, επέλεξα το **Transform Cases (lower case)** γιατί ήθελα να μετατρέψω όλα τα γράμματα μου σε πεζά.

Στην επεξεργασία των δεδομένων για κάθε κατηγορία (TF-IDF, Term Occurrences, Binary Term Occurrences) αντίστοιχα έτρεξα 4 διαφορετικά πειράματα για διαφορετικές τιμές του prune. Δηλαδή **3(κατηγορίες) x 4(διαφορετικά ποσοστά prune) = 12 πειράματα**. Παρακάτω υπάρχουν όλα τα screenshots από τα αποτελέσματα που παράχθηκαν.

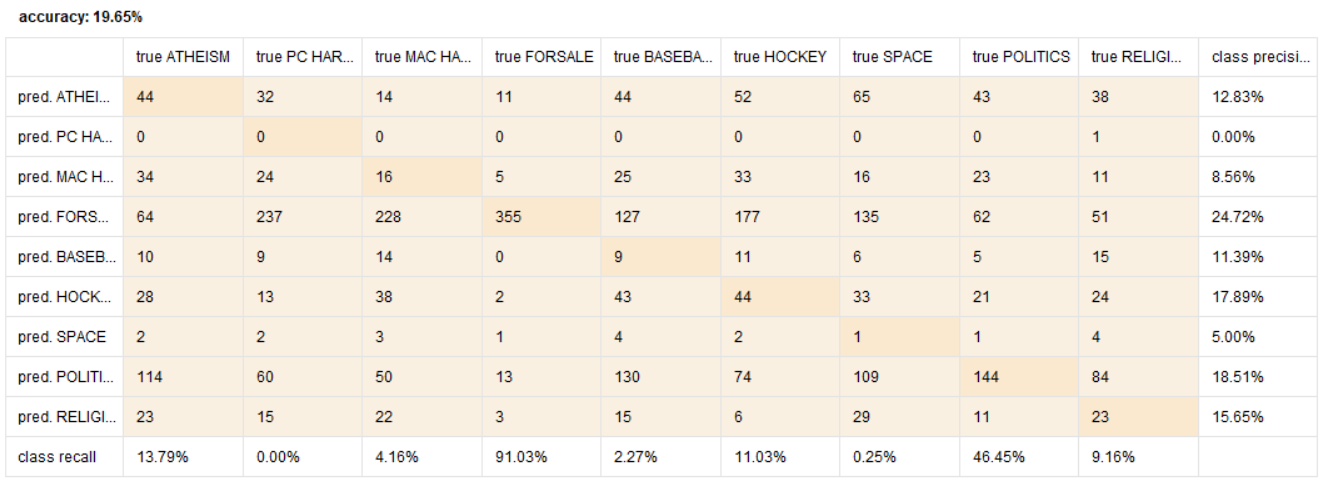
**Vector creation: TF-IDF, prune below: 10%, prune above: 80%**



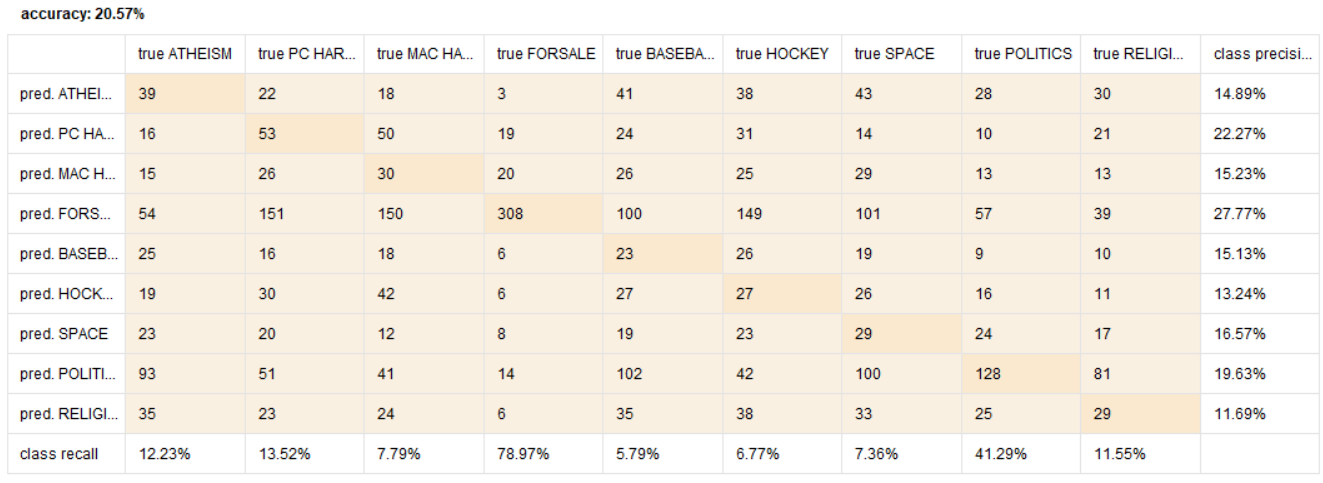
**Vector creation: TF-IDF, prune below: 15%, prune above: 90%**



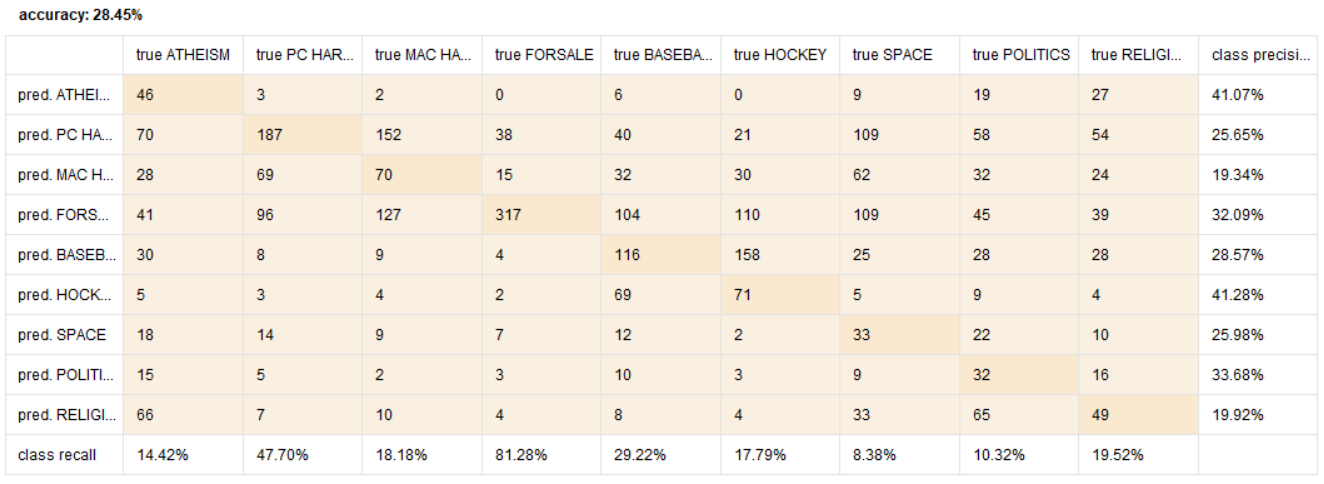
**Vector creation: TF-IDF, prune below: 30%, prune above: 70%**



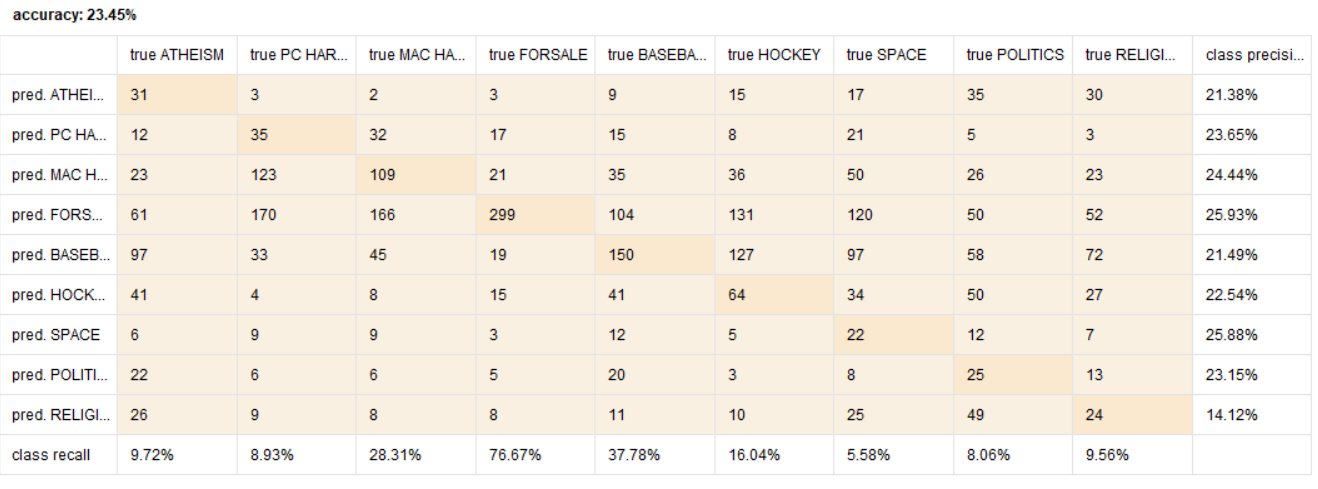
**Vector creation: TF-IDF, prune below: 25%, prune above: 85%**



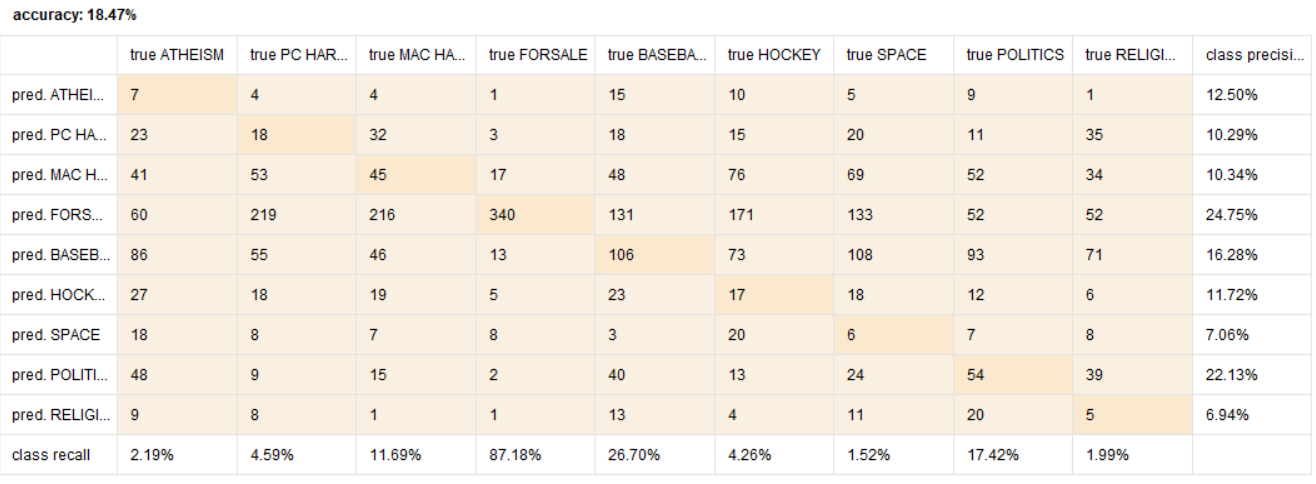
**Vector creation: Term Occurrences, prune below: 10%, prune above: 80%**



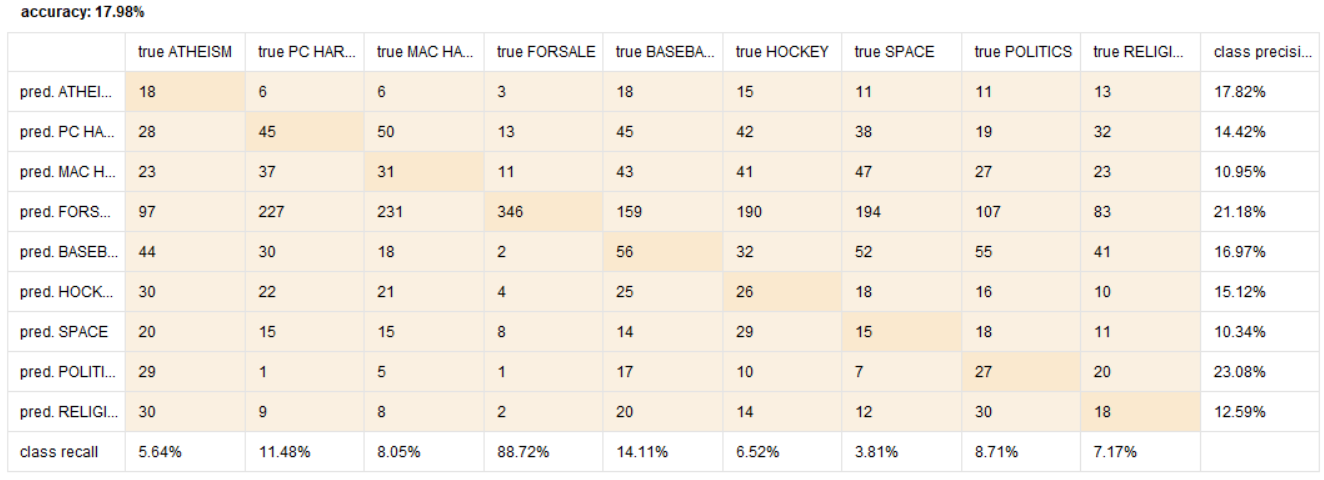
**Vector creation: Term Occurrences, prune below: 15%, prune above: 90%**



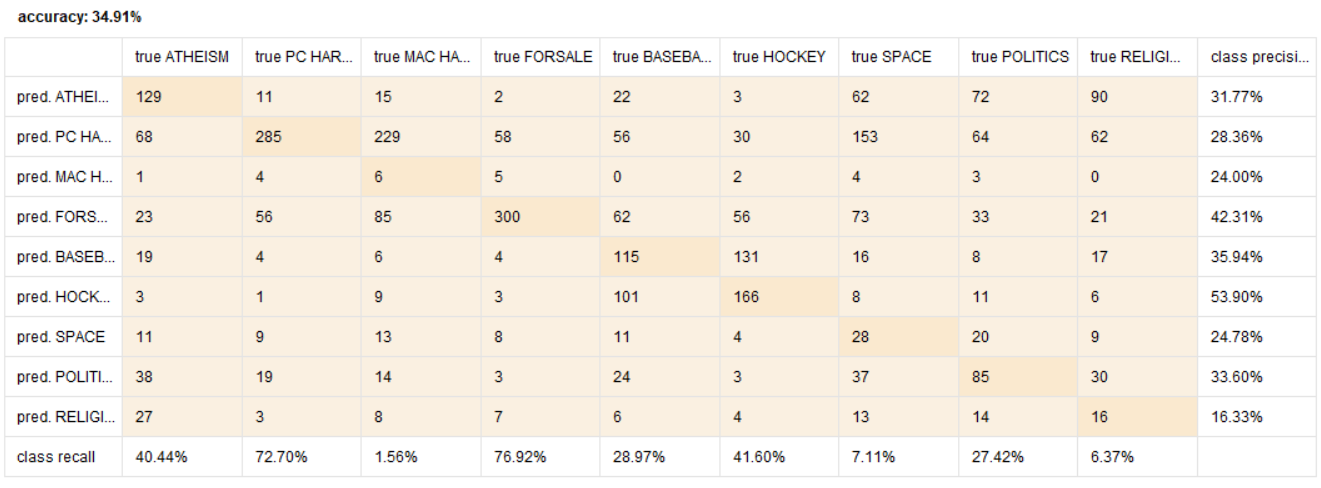
**Vector creation: Term Occurrences, prune below: 30%, prune above: 70%**



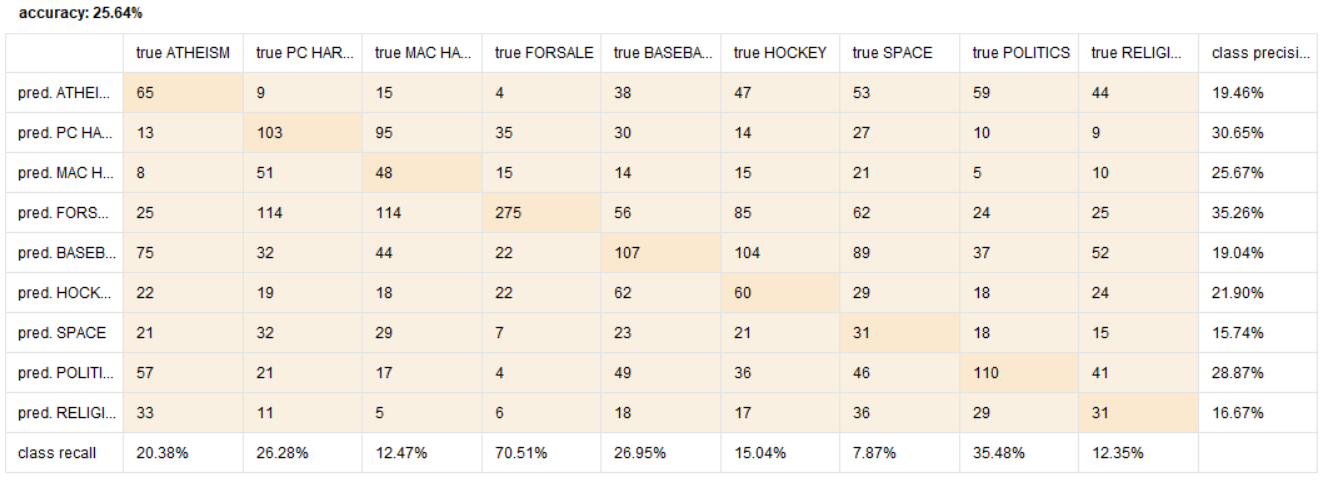
**Vector creation: Term Occurrences, prune below: 25%, prune above: 85%**



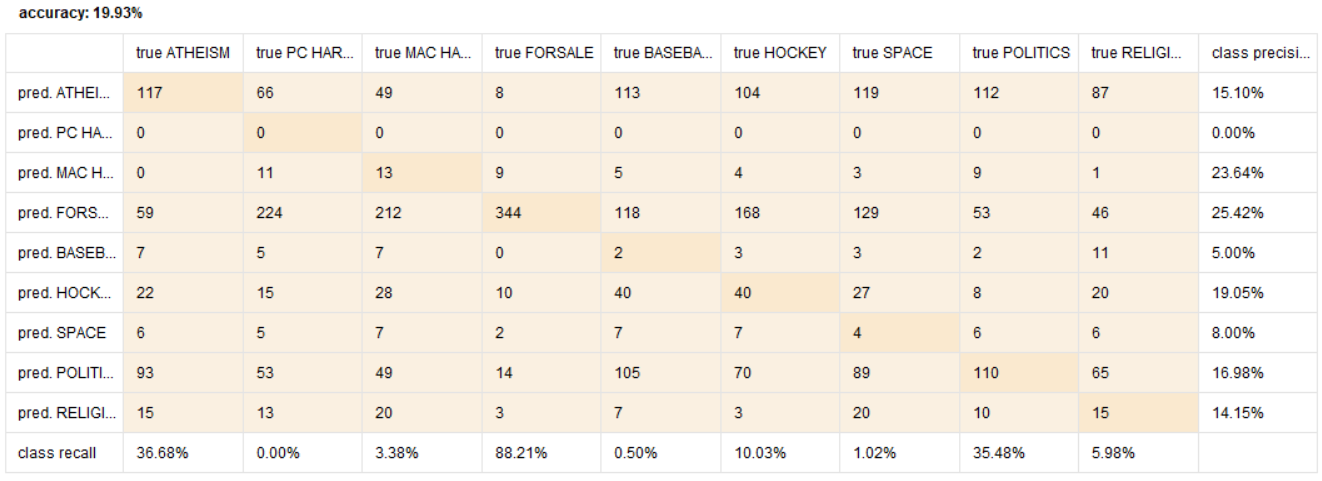
**Vector creation: Binary Term Occurrences, prune below: 10%, prune above: 80%**



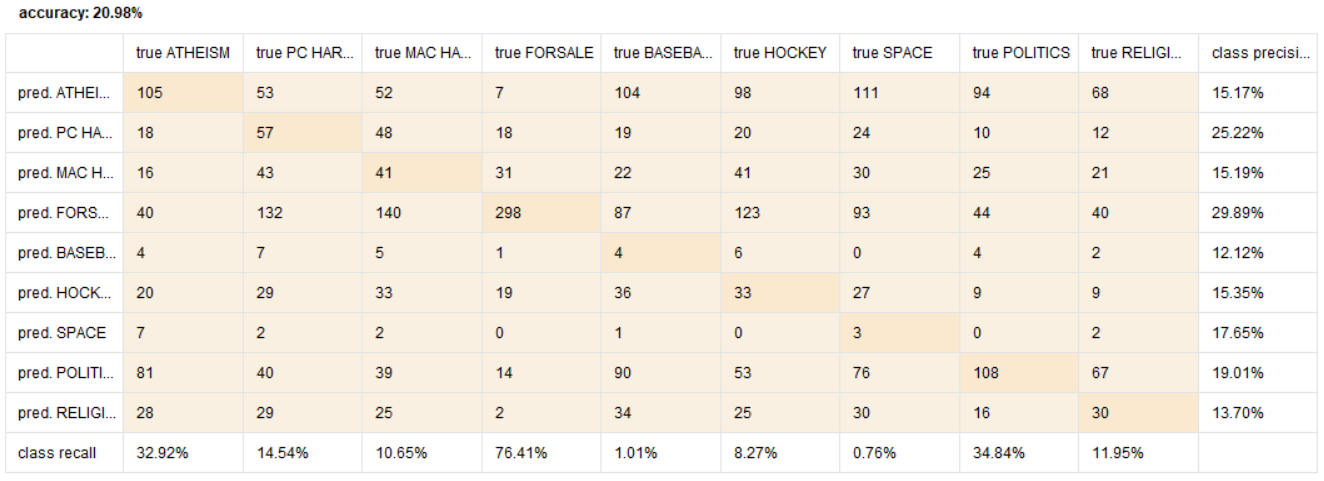
**Vector creation: Binary Term Occurrences, prune below: 15%, prune above: 90%**



**Vector creation: Binary Term Occurrences, prune below: 30%, prune above: 70%**



**Vector creation: Binary Term Occurrences, prune below: 25%, prune above: 85%**



**ΣΥΜΠΕΡΑΣΜΑ:** Σε κάθε κατηγορία (TF-IDF, Term Occurrences, Binary Term Occurrences) το μεγαλύτερο accuracy το συναντήσαμε στο prune (below = 10% και above = 80%) με ποσοστά ( 37.66%, 28.45%, 34.91%) αντίστοιχα. Άρα μας βολεύει να κόβουμε τις λέξεις που εμφανίζονται σε ποσοστό κάτω από 10% των εγγράφων και επίσης να κόβουμε τις πολύ συνηθισμένες λέξεις.

**ΑΠΟ ΤΟΝ ΦΟΙΤΗΤΗ: Τσιαούση Σταύρο (dai17173)**